

# VBA: Open Google Maps From ArcMap to the Address of Selected Table Record

Contributed by Bert Granberg  
23, Dec. 2009  
Last Updated 23, Dec. 2009

In attempt at resourcefulness, if I can't find geocode an address against the SGID93.Transportation.Streets dataset or the web services that run against this same dataset, I'll check to see if Google knows where it is.

This script (below) will automatically open a browser window and attempt to find and zoom the Google map view to a an address specified in a selected ArcMap table record. You'll just need to configure a custom button in ArcMap in order to make this easier to use and set 3 parameters within the script itself:

- Table position in the table of contents (usually 0);
- the fieldname for the address attribute field; and
- the fieldname for the zone attribute field (usually zipcode).

Here's how

## Part I. Paste code into VBA Editor

- 1) From the ArcMap menu, navigate to Tools --> Macros --> Visual Basic Editor to open the VBA editor
- 2) In the VBA Editor's Project Window, expand the Project --> Arcmap Objects folder and double click on ThisDocument
- 3) Paste the code (below) in the window that opens and close the VBA editor. Important: Make sure that the first 10 or so lines of code (before the Public Sub line) precede any procedures in your code window and are not already in the code window from the Google StreetView script (no need to duplicate them).

- 4) Save your project

## Part II. Create and configure custom ArcMap Tool

- 1) From the ArcMap menu, navigate to Tools -->Customize
- 2) Click the Commands tab.
- 3) Click the drop-down arrow on the Save in combo box, then change the selected item so it is the current name of your ArcMap project (ex. Untitled.mxd)
- 4) Click [UIControls] in the Categories list. then Click the New UIControl button.
- 5) Click the UIToolControl radio button and then click Create.
- 6) The name of the new tool appears in the Commands list. Single click the newly created UIControl, then click it again to activate in-place editing, then type a new name for the control so it reads Project.GoogleMap\_GeocodeLink.
- 7) Click and drag the newly created UIControl and drop it on an ArcMap toolbar or menu.
- 8) On the toolbar or menu, right-click the command to set its image, caption, and other properties.
- 9) Save Project and start using the tool.

## Option Explicit

```
Const SW_SHOWMAXIMIZED = 3  
Const SW_SHOWMINIMIZED = 2  
Const SW_SHOWDEFAULT = 10
```

```
Const SW_SHOWMINNOACTIVE = 7
Const SW_SHOWNORMAL = 1
```

```
Private Declare Function ShellExecute Lib "shell32.dll" _
Alias "ShellExecuteA" (ByVal hWnd As Long, _
ByVal lpOperation As String, ByVal lpFile As String, _
ByVal lpParameters As String, ByVal lpDirectory As String, _
ByVal nShowCmd As Long) As Long
```

'-----everything above this line must go before any procedures in your code window

```
Public Sub GoogleMap_GeocodeLink()
```

```
    Dim tablePosition As Integer
    Dim addressFieldName As String
    Dim zipFieldName As String
```

```
    '**** SET THESE PARAMETERS
    tablePosition = 0
    addressFieldName = "Address"
    zipFieldName = "Zip"
    '****
```

```
    Dim pMxD As IMxDocument
    Dim pSATC As IStandaloneTableCollection
    Dim pSATable As ITable
    Dim pDataset As IDataset
    Dim pTableSel As ITableSelection
    Dim pRow As IRow
    Dim addressFieldIndex As Integer
    Dim zipFieldIndex As Integer
    Dim addressStr As String
    Dim zoneStr As String
    Dim URLStr As String
    Dim openBrowserReturn As Long
```

```
    Set pMxD = ThisDocument
    Set pSATC = pMxD.FocusMap
    Set pSATable = pSATC.StandaloneTable(0)
    Set pDataset = pSATable
    Set pTableSel = pSATable
```

```
    If pTableSel.SelectionSet.Count <> 1 Then
```

```
        MsgBox "One and only one record must be selected in table: " & pDataset.Name
    Else
```

```
        Set pRow = pSATable.GetRow(pTableSel.SelectionSet.IDs.Next)
```

```
        addressFieldIndex = pSATable.FindField(addressFieldName)
        zipFieldIndex = pSATable.FindField(zipFieldName)
```

```
        If addressFieldIndex < 0 Or zipFieldIndex < 0 Then
```

```
            MsgBox "Field name not found for zip or address field, check VBA script settings"
            Exit Sub
        End If
```

```
        addressStr = pRow.Value(addressFieldIndex)
        zoneStr = pRow.Value(zipFieldIndex)
        URLStr = "http://maps.google.com/maps?f=q&source=s_q&hl=en&geocode=&q="
        URLStr = URLStr & Replace(addressStr, " ", "+") & "," & zoneStr
```

```
        openBrowserReturn = OpenLocation(URLStr, SW_SHOWNORMAL)
    End If
```

```
End Sub
```

Public Function OpenLocation(URL As String, WinState As Long) As Long

'PURPOSE: Opens default browser to display URL

'RETURNS: module handle to executed application or  
'Error Code ( < 32) if there is an error

'can also be used to open any document associated with  
'an application on the system (e.g., passing the name  
'of a file with a .doc extension will open that file in Word)

Dim IHWnd As Long  
Dim IAns As Long

IAns = ShellExecute(IHWnd, "open", URL, vbNullString, \_  
vbNullString, WinState)

OpenLocation = IAns

End Function